Sheet 3

1) suppose client "A" initiates a Telenet session with server "S". At about the same time, client "B" initiates a Telenet session with server "S". Provide source and destination Port numbers for.

a) segments sent from A to S.

src = 2000 (random)

dst = 23 (telenet)

b) segments sent from B to S

src = 3555 (random)

dst = 23

c) segments sent from S to A

src = 23 (telenet) , dst = 2000 (random)

d) segments sent from S to B

src = 23 (telenet) , dst = 3555 (random)

e) if A, B different hosts, it is Possible that src. Port number in segments from A to S is same with segments sent from B to S?

↳ It is Possible for different processors in different hosts to have same Port number.

f) what about if it is the same host?

↳ It is not Possible.

[3] UDP & TCP use 1's complement for their checksums. Suppose you have the following three 8-bit bytes 01010011, 01100110, 01110100

a) what is the 1's complement of sum of them (Note that TCP use 16-bit words in computing checksum, for this Problem you are being asked to use 8-bit sums?

```
  1 1   1
  0 1 1 1 0 1 0 0
  0 1 1 0 0 1 1 0
  0 1 0 1 0 0 1 1
-----------------
  0 0 1 0 1 1 0 1      Sum
-----------------
  1 1 0 1 0 0 1 0      1's Complement
```

b) why UDP uses 1's Comp. ?

↳ to Add 1's complement to sum of data to check errors.

c) Is it possible that 1-bit error will go undetected?

↳ No

d) How about 2-bit error?

↳ It is possible.

26 Consider transferring an enormous file of L bytes from host A to host B, Assume an MSS of 536 bytes.

a. What is the maximum value of L such that TCP sequence numbers are not exhausted? recall that TCP sequence number field has 4 bytes.

maximum L = 2^{32} Byte (Packet size)

b. For L in (a), find how long it takes to transmit the file. Assume that 66 bytes of transport, network and data link header are added to each segment before resulting Packet is sent out over 155 Mbs link (ignore flow and congestion control)

no. of segments = $\frac{2^{32}\ B}{536\ B}$ = 8012999 segments

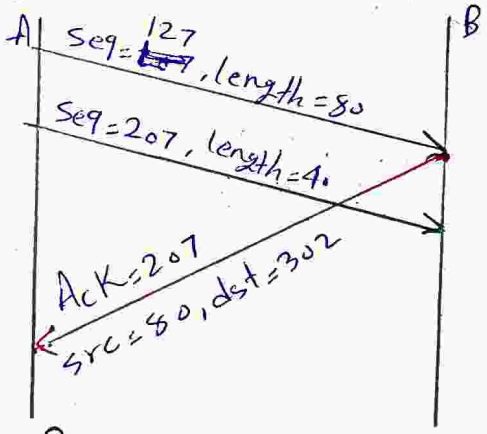
Total size of header = 66 * 8012999 = 528857913 Byte

Total size of data = 2^{32} + 528857913 = 4823825209 Byte

Time = $\frac{8 * 4823825209}{155 * 10^{6}}$ = 249 sec

27 Host A, B communicating over TCP connection, and host B already received from A all bytes up through byte 126. Suppose host A then sends two segments to host B back-to-back. First and second segments contain 80 and 40 bytes of data respectivly. 1st segment (sequence number = 127, src. port number = 302, dst. port number = 80) Host B sends acknowledgment whenever it receives segment from Host A.

a) in 2nd segment sent from A to B, what are the sequence number, src. port number and dst. port no.?



sequence no. = 207

src port no. = 302

dst. port no. = 80

b) if first segment arrives before 2nd segment, in the acknowledgment of first arriving segment? what is

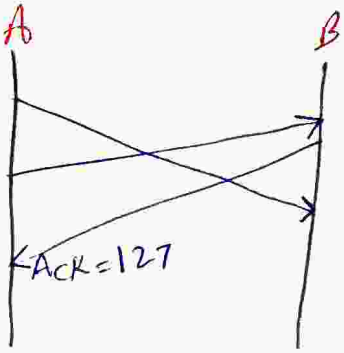
Acknowledge no. = 207

src. port no. = 80

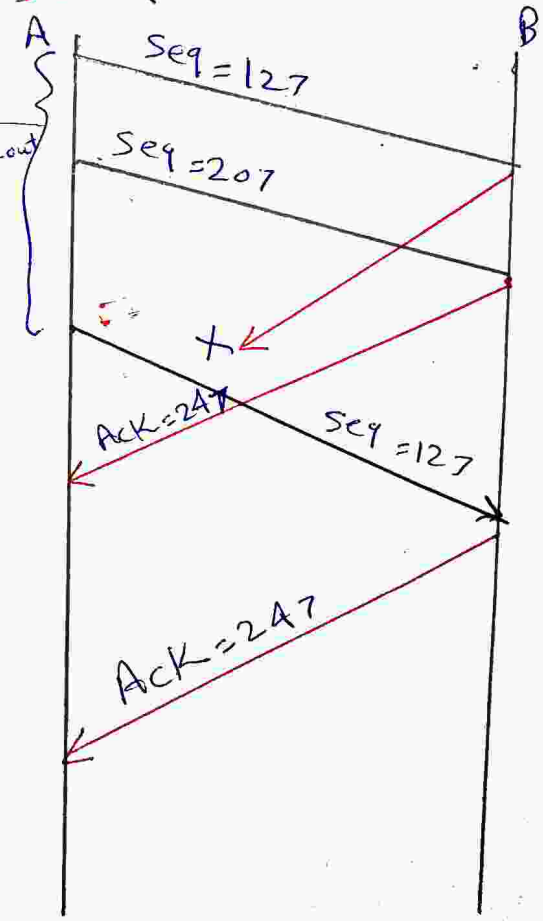
dst port no. = 302

c) if 2nd segment arrives before 1st one, in the acknowledgment of first arriving segment what is the acknowledgment number?

Ack no. = 127



d) suppose the two segments sent by A arrive in order at B. First Ack. is lost and 2nd Acknowledgment arrives after first time-out interval Draw training diagram show these segments and all other segments sent (Assume there is no additional Packet loss) For each segment, Provide sequence number and no. of bytes of data, for each Ack. that you add, Provide ack. number



[31] Suppose that the five measured sample RTT values (section 3.5.3 refrence) are 106 ms, 120 ms, 140 ms, 90 ms and 115 ms a) Compute estimated RTT after each of these sample RTT values, using $\alpha = 0.125$ and assume that estimated RTT was 100 ms just before first of five samples were obtained?

Estimated RTT = $(1-\alpha)$ Estimated RTT + α sample RTT

1st value → ETT = $0.875 * 100 + 0.125 * 106 = 100.75$ ms

2nd value → ETT = $0.875 \times 100.75 + 0.125 * 120 = 103.15$ ms

3rd → 107.76 ms & 4th → 105.54 ms

5th → 106.72 ms

b) Compute also DevRTT after each sample obtained assume $\beta = 0.25$ (value of DevRTT was 5 ms & Compute TCP timeout interval after each of samples obtained.

DevRTT = $(1-\beta) \cdot$ DevRTT + β |SampleRTT − Estimated RTT|

Timeout interval = Estimated RTT + 4 * DevRTT

	DevRTT	Timeout-Interval
1st →	5.0625 ms	121 ms
2nd →	8 ms	135.15 ms
3rd →	14.06 ms	164 ms
4th →	14.43 ms	163.26 ms
5th →	12.8925 ms	158.29 ms